Robust Methods for Real-Time Diabetic Foot Ulcer Detection and Localization on Mobile Devices

Manu Goyal, *Student Member, IEEE*, Neil D. Reeves, Satyan Rajbhandari, and Moi Hoon Yap, *Member, IEEE*

***Abstract*—Current practice for diabetic foot ulcers (DFU) screening involves detection and localization by podiatrists. Existing automated solutions either focus on segmentation or classification. In this work, we design deep learning methods for real-time DFU localization. To produce a robust deep learning model, we collected an extensive database of 1775 images of DFU. Two medical experts produced the ground truths of this data set by outlining the region of interest of DFU with an annotator software. Using five-fold cross-validation, overall, faster R-CNN with InceptionV2 model using two-tier transfer learning achieved a mean average precision of 91.8%, the speed of 48 ms for inferencing a single image and with a model size of 57.2 MB. To demonstrate the robustness and practicality of our solution to real-time prediction, we evaluated the performance of the models on a NVIDIA Jetson TX2 and a smartphone app. This work demonstrates the capability of deep learning in real-time localization of DFU, which can be further improved with a more extensive data set.**

***Index Terms*—Diabetic foot ulcers, deep learning, convolutional neural networks, DFU localization, real-time localization.**

I. Introduction

DIABETIC foot ulcers (DFU) that affect the lower extremities are a major complication of Diabetes. According to the global prevalence data of International Diabetes Federation in 2015, annually, DFU develop in 9.1 million to 26.1 million people with diabetes worldwide [1]. It has been estimated that patients with diabetes have a lifetime risk of 15% to 25% in developing a DFU, with 85% of lower limb amputations occurring due to an infected DFU that did not heal [2], [3]. In a more recent study, when additional data is considered, the risk is suggested to be in-between 19% to 34% [4].

Due to the proliferation of Information Communication Technology, the intelligent automated telemedicine systems are often tipped as one of the most cost-effective solutions for remote detection and prevention of DFU. Telemedicine systems along with current healthcare services can integrate with each other to provide more cost-effective, efficient and quality treatment for DFU. In recent years, there has been a rapid development in computer vision, especially towards the difficult and vital issues of understanding images from different domains such as spectral, medical, object detection [5] and human motion analysis [6]. The computer vision and deep learning algorithms are extensively used for the analysis of medical imaging of various modalities such as MRI, CT scan, X-ray, dermoscopy, and ultrasound [7]. Recently, computer vision algorithms are extended to assess different types of skin condition such as skin cancer and DFU [8], [9].

From a computer vision and medical imaging perspective, there are three common tasks that can be performed for the detection of abnormalities on medical images, which are 1) Classification 2) Localization 3) Segmentation. These tasks on DFU are illustrated by Fig. 1. Various researchers have made contributions related to computerised methods for the detection of DFU. We divided these contributions into four categories:

1) Algorithms development based on basic image processing and traditional machine learning techniques
2) Algorithms development based on deep learning techniques
3) Research based on different modalities of images
4) Smartphone applications for DFU

Several studies suggested computer vision methods based on basic image processing approaches and supervised traditional machine learning for the detection of DFU/wound. Mainly, these studies have performed the segmentation task by extracting texture descriptors and color descriptors on small patches of wound/DFU images, followed by traditional machine learning algorithms to classify them into normal and abnormal skin patches [11]–[14]. In conventional machine learning, the handcrafted features are usually affected by skin shades, illumination, and image resolution. Also, these techniques struggled to segment the irregular contour of the ulcers or wounds. On the other hand, the unsupervised approaches rely upon image

Manuscript received March 14, 2018; revised June 3, 2018 and August 7, 2018; accepted August 30, 2018. Date of publication October 4, 2018; date of current version July 1, 2019. *(Corresponding author: Moi Hoon Yap.)*

M. Goyal and M. H. Yap are with the School of Computing, Mathematics and Digital Technology, Manchester Metropolitan University, Manchester M1 5GD, U.K. (e-mail: manu.goyal@stu.mmu.ac.uk; m.yap@mmu.ac.uk).

N. D. Reeves is a Professor of musculoskeletal biomechanics with the Research Centre for Musculoskeletal Science & Sports Medicine, School of Healthcare Science, Faculty of Science and Engineering, Manchester Metropolitan University, Manchester M1 5GD, U.K. (e-mail: n.reeves@mmu.ac.uk).

S. Rajbhandari is with Lancashire Teaching Hospital, Preston PR2 9HT, U.K. (e-mail: Satyan.Rajbhandari@lthtr.nhs.uk).

Digital Object Identifier 10.1109/JBHI.2018.2868656

Fig. 1. Examples of three common tasks for inspection of abnormalities on a DFU image. (a) Classification, (b) localization, and (c) segmentation of DFU (green) and surrounding skin (red) [10].

processing techniques, edge detection, morphological operations and clustering algorithms using different color space to segment the wounds from images [15]–[17]. Wang *et al.* [18] used an image capture box to capture image data and determined the area of DFU using cascaded two-stage SVM-based classification. They proposed the use of superpixel technique for segmentation and extracted the number of features to perform two-stage classification. Although this system reported promising results, it has not been validated on a more substantial dataset. In addition, the image capture box is very impractical for data collection as there is a need for the patient's barefoot to be placed directly in contact with the screen of image capture box. In healthcare, such setting would not be allowed due to the concerns regarding infection control.

The majority of these methods involve manually tuning of the parameters according to different input images and multi-stage processing which make them hard to implement in clinical settings. These state-of-the-art methods were validated on relatively small datasets, ranging from 10 to 172 images. Current state-of-the-art methods based on basic image processing and traditional machine learning techniques are not robust, due to their nature of reliance on specific regulators and rules, with certain assumptions.

In contrast to traditional machine learning, deep learning methods do not require such intense assumptions and have demonstrated superiority in DFU localization and segmentation of DFU, which suggests that the robust fully automated detection of DFU may be achieved, by adopting such approach [9], [10], [19]. In the field of deep learning, several researchers made contributions on the classification and segmentation of DFU. Goyal *et al.* [9] proposed a new deep learning framework called DFUNet which classified the skin lesions of the foot region into two classes, i.e. normal skin (healthy skin) and abnormal skin (DFU). In addition, they used deep learning methods for the semantic segmentation of DFU and its surrounding skin with a limited dataset of 600 images [10]. Wang *et al.* [19] proposed a new deep learning architecture based on encoder-decoder to perform wound segmentation and analysis to measure the healing progress of wound. To date, this paper is the first attempt to develop deep learning methods for the DFU localization task.

Then, in a separate study from computer vision techniques, van Netten *et al.* [20] proposed the detection of DFU using a different modality called infra-red thermal imaging. They found that there is a significant temperature difference between the DFU and the surrounding healthy skin of the foot. Hence, they used this considerable temperature difference on a heat-map to detect the DFU. Liu *et al.* presented a preliminary case study to evaluate the effectiveness of infra-red dermal thermography on diabetic feet soles to identify pre-signs of ulceration [21]. Harding *et al.* [22] performed a study to assess the infra-red imaging for the prevention of secondary osteomyelitis. Similarly, infra-red thermography has been used in various studies to detect the complications related to the DFU [23], [24].

Health applications on the smartphone are fast becoming popular in monitoring essential aspects of the human body. Yap *et al.* [25], [26] developed an app called FootSnap, which is used to produce the standardized dataset of the DFU images. This application used basic image processing techniques such as edge detection to provide the ghost images of the foot which is useful to monitor the progress of DFU. Since this was designed to standardise image capture conditions, it did not perform any automated detection function. Recently, Brown *et al.* [27] developed a smartphone application called MyFootCare, which provides useful guidance to the DFU patients as well as keep the record of foot images. In this application, the end-users need to crop the patch of the captured image, and with basic color clustering algorithms, it can produce DFU segmentation. But, previous research [10] has already shown that the basic clustering algorithms are not robust enough to provide accurate DFU segmentation on full foot images.

The major challenges of DFU localization task are as follow: 1) Expensive in data collection and expert labelling on the DFU dataset; 2) High inter-class similarity between the DFU lesions and intraclass variation depending upon the classification of DFU [29]; and 3) Lighting conditions and patient's ethnicity. In this work, we provide a large-scale annotated DFU dataset and propose an end-to-end mobile solution for DFU localisation. The key contributions of this paper include:

1) We present one of the largest DFU dataset, which consists of 1775 images with annotated bounding box indicating the ground truth of DFU location. To date, the largest dataset we encountered is of 600 DFU images, where it was used for the semantic segmentation of DFU and its surrounding skin [10].
2) We propose the use of convolutional neural networks (CNNs) to localize DFU in real-time with two-tier transfer learning. To our best knowledge, this is the first time CNNs are used for this task. Since our main focus is

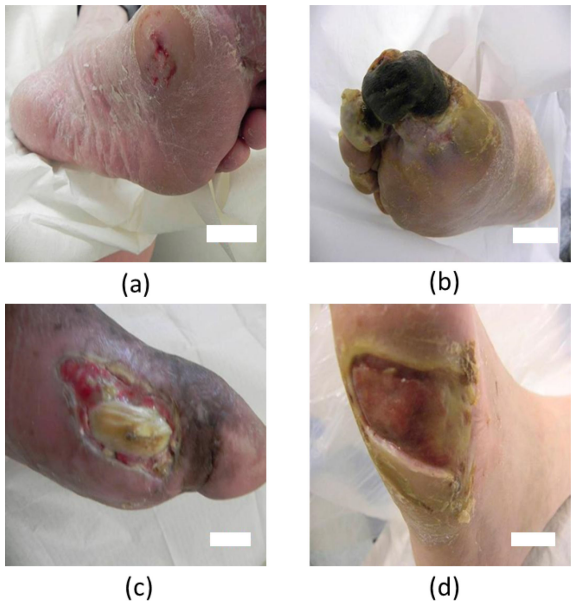

Fig. 2. Illustration of high-resolution full feet images of our DFU dataset.

on mobile devices, we emphasize on light-weight object localization models.

3) Finally, we demonstrate the application of our proposed methods on two types of mobile devices: Nvidia Jetson TX2 and an android mobile application.

II. Methodology

This section describes the preparation of the dataset and expert labeling of the DFU on foot images. The description of CNNs for DFU localization is detailed. Finally, the performance metrics used for validation are reported.

A. DFU Dataset

We received NHS Research Ethics Committee approval with REC reference number 15/NW/0539 to use the foot images of DFU for our research. Foot images with DFU were collected from the Lancashire Teaching Hospitals over the past few years. Our dataset has a total of 1775 foot images with DFU. There were three cameras mainly used for capturing the foot images, Kodak DX4530, Nikon D3300 and Nikon COOLPIX P100. Whenever possible, the images were acquired with close-ups of the full foot with the distance of around 30–40 cm with the parallel orientation to the plane of an ulcer. The use of flash as the primary light source was avoided, and instead, adequate room lights are used to get the consistent colors in images. The sample foot images in the dataset are shown in the Fig. 2. To test the specificity measure for the algorithms, we have included 105 healthy foot images in the DFU dataset from the FootSnap application [26].

In this dataset, the size of images varies between 1600×1200 and 3648×2736. We resized all the images to 640×640 to improve the performance and reduce the computational costs. We used Hewitt *et al.* [28] annotation tool for producing the ground truths in the form of bounding box as shown in Fig. 3.

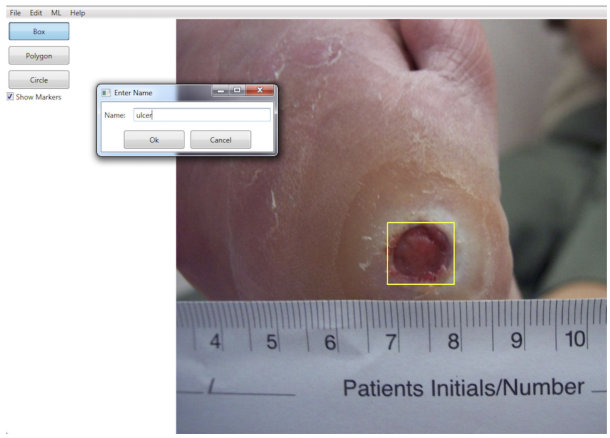

Fig. 3. Example of delineating ground truth on DFU dataset using Brett *et al.* annotation tool [28].

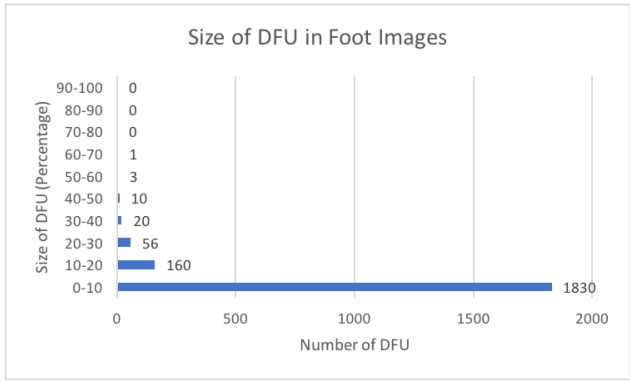

Fig. 4. Comparison of Size of DFU against the size of image.

The ground truth was produced by two healthcare professionals (a podiatrist and a consultant physician with specialization in the diabetic foot) specialized in diabetic wounds and ulcers. When there was disagreement, the final decision was mutually settled with the consent of both. In the DFU dataset, there is only one bounding box in approximately 90% of the images, two bounding boxes in 7% and finally, more than two bounding boxes in the remaining 3% images of the whole dataset. The medical experts delineated a total of 2080 DFUs (some images with more than one ulcer) using an annotator software. As shown in the Fig. 4, approximately 88% DFU have the size less than 10% of the actual size of an image. The size varied considerably across the DFUs in the dataset.

B. Conventional Methods for DFU Localization

In this section, we assessed the performance of conventional methods for the localization of DFU. For traditional machine learning, we delineated 2028 normal skin patches and 2080 abnormal skin patches for feature extraction and training of classifier using 5-fold cross-validation [9]. We also used data-augmentation techniques such as flipping, rotation, random crop, color channels to make a total of 28392 normal and 29120 abnormal patches. 80% of the image data is used to train the classifier and remaining 20% of the data is used as test im-

ages. Since these two classes of skin (normal and abnormal) have significant textural differences amongst them, we investigated various feature extraction techniques including low-level features such as edge detection, corner detection [30], texture descriptors such as Local Binary Patterns (LBP) [31], Gabor filter [32], Histogram of Oriented Gradients (HOG) [33], shape based descriptors such as hough transform [34] and color descriptors such as Normalized *RGB, HSV, and L*u*v* features [35]. With exhaustive feature selection technique, we settled with LBP, HOG, color descriptors to extract features from skin patches of both normal and abnormal classes. For a single patch, 209 features were extracted with above mentioned feature extraction techniques. After the feature extraction from images, we used Quadratic support vector machine [36] as a classifier for the classification task. Then, to perform DFU localization task with multiple scales, we used the sliding window approach to mask each box if the corresponding patch is detected as ulcer by trained classifier.

This technique has achieved a good score in evaluation metrics, 70.3% in *Mean Average Precision*. The conventional machine learning methods require a lot of intermediate steps like pre-processing of images, extracting hand-crafted features and multiple stages to get the final results which makes them very slow. Whereas, deep learning provides the faster end-to-end models on various computing platforms which simply take images as input and provide the final localization results as output.

C. Deep Learning Methods for DFU Localization

CNNs proved their superiority compared to the conventional machine learning techniques in image recognition tasks such as ImageNet [37] and MS-COCO challenges [38]. They are very capable of classifying the images into different classes of objects from both non-medical and medical imaging by extracting the hierarchies of features. One of the important tasks in computer vision is object localization where algorithms need to localize and identify the multiple objects in an image. Mainly, object localization networks consist of three stages as described in the following subsections.

1) CNN as Feature Extractor: In Stage 1, the standard CNN such as MobileNet, InceptionV2, the convolutional layers extract the features from input images as feature maps. These feature maps are used to identify the objects in the image with particular attention focused on DFU regions as shown in the Fig. 5. These feature maps serve as input for the later stages such as generation of proposals in the second stage and classification and regression of RoI in the third stage.

2) Generation of Proposals and Refinement: In Stage 2, the network scans the image in a sliding-window fashion and finds specific areas that contain the objects using the feature map extracted in Stage 1. These areas are known as proposals which have different boxes distributed over the image. In general, around 200,000 proposals of different sizes and aspect ratios are found to cover as many objects as possible in the image. With GPU, Faster-RCNN produces these much anchors in 10 ms [39]. Stage 2 generates two outputs for each proposal:

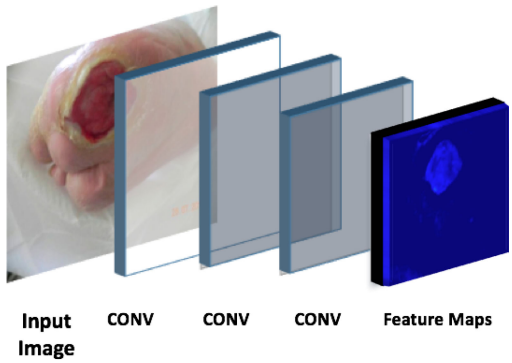

Fig. 5. Stage 1: The feature map extracted by CNN that acts as backbone for object localization network. Conv refers convolutional layer.

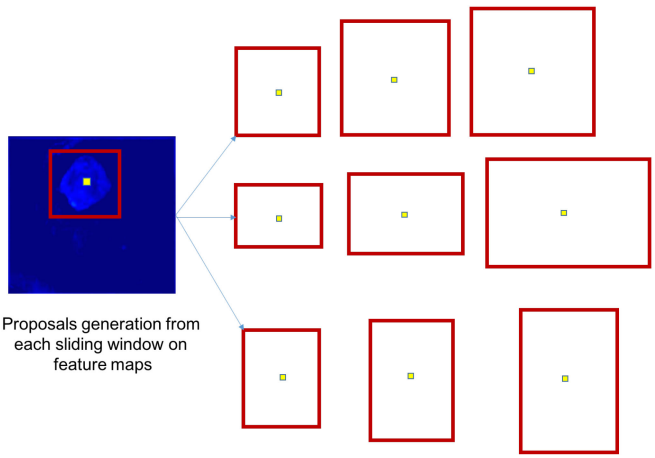

Fig. 6. Stage 2: Detected proposal boxes with translate/scale operation to fit the object. There can be several proposals on a single object.

- Proposal Class: It can be either foreground or background. The foreground class means there is likely an object in that proposal and it is also known as a positive proposal.
- Proposal Refinement: A positive proposal might not be perfectly captured the object. So the network estimates a delta (% change in x, y, width, height) for refinement of the proposal box to center the object better as illustrated in Fig. 6.

3) RoI Classifier and Bounding Box Regressor: Stage 3 consists of the classification of RoI boxes provided by Stage 2 and further refinement of the RoI boxes as shown in the Fig. 7. First, all RoI boxes are fed into the RoI pooling layer to resize them into fixed input size for classifier as RoI boxes can have different sizes. Similar to Stage 2, it generates two outputs for each RoI:

- RoI Class: The softmax layer provides the classification of regions to specific classes (if more than one class). If the RoI is classified as background class, it is discarded.
- Bbox Refinement: Its purpose is to refine the location of RoI boxes.

We considered three types of object localization networks to perform on the DFU dataset. First is Faster R-CNN [39], which is a successor of Fast R-CNN [40] for object localization in terms of speed. It consists of all three stages of object localization network as shown in the Fig. 8. It has two-stage loss

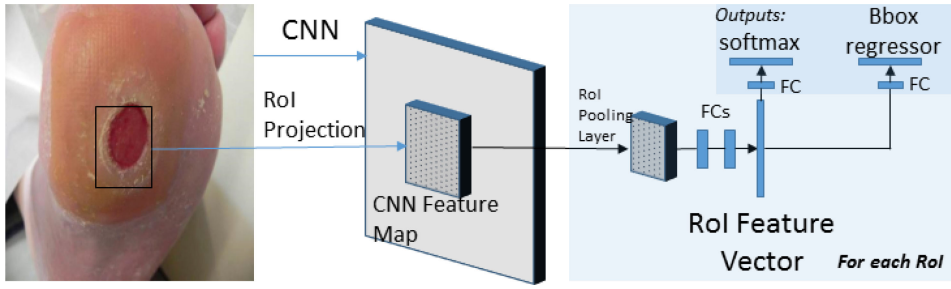

Fig. 7. Illustration of Stage 3: The classification and further box refinement of RoI boxes from the second stage proposal with softmax and Bbox regression. Where FC refers to Fully-connected layer.

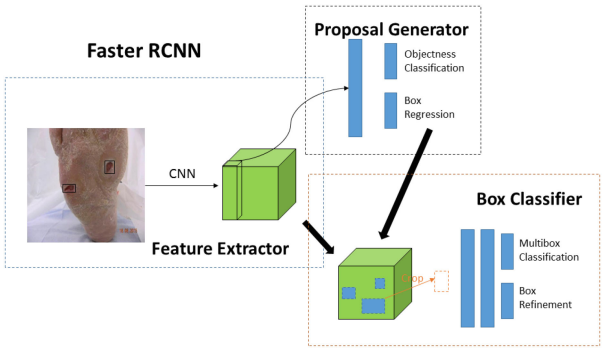

Fig. 8. Faster R-CNN architecture for DFU localization which consists of all three stages discussed earlier.

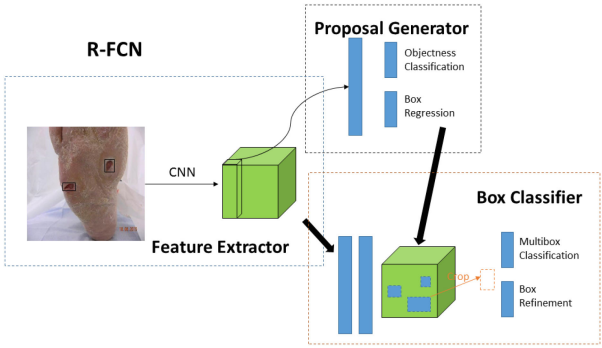

Fig. 9. R-FCN architecture which considers only the feature map from the last convolutional layer which speeds up the three stage network.

function whereas first stage loss function that consists of the parameters such as space, scale and aspect ratio of the proposals. Then, second stage loss function re-runs the crops of proposal produced by the second stage with feature extractor to produce more accurate box proposals for classification.

Dai *et al.* [41] proposed the Region-based Fully Convolutional Networks (R-FCN) to produce faster box proposals by considering the crops only from the last layer of features with comparable accuracy as Faster R-CNN which crop features from the same layer where region proposals are predicted as shown in the Fig. 9. Due to cropping limited only to the last layer, it minimizes the time to get the box refinement.

Single Shot Multibox Detector (SSD) [42] is a new architecture for the object localization which uses a single stage CNN to predict classes directly and anchor offsets without the need of

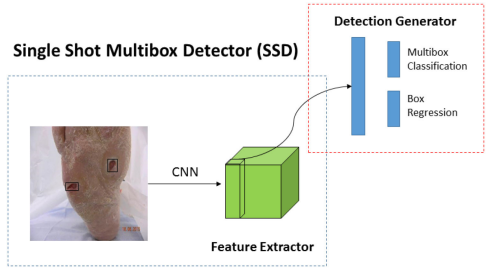

Fig. 10. The architecture of Single Shot Multibox Detector (SSD). It considers only two stage by eliminating the last stage to produce faster box proposals.

second stage proposal generator unlike Faster R-CNN [39] and R-FCN [41] as shown in the Fig. 10. The SSD meta-architecture produces anchors much faster than other object localization networks, which makes it more suitable for the mobile platforms.

There are six popular state-of-the-art object localization models which are based on these three region based detector meta-architectures i.e. Single Shot multibox detector [42], R-FCN [41] and Faster R-CNN [39]. These three meta-architectures used the state-of-the-art classification algorithms like MobileNet [43], InceptionV2 [44], ResNet101 [45], Inception-ResNetV2 [46] to get the anchor boxes from the features maps, and finally, classify these anchors to different classes. Table I summarises the size of models, speed (inference per image), and accuracy (mAP) trained on MS-COCO dataset with 90 classes [38], [47].

Since our work is limited by the hardware on mobile devices and real-time prediction, we only considered lightweight models (very small, low latency) in terms of size of the model and inference speed. We used the first three models (SSD-MobileNet, SSD-InceptionV2 and Faster R-CNN with InceptionV2) for the DFU dataset as illustrated in Table I. These small models are specifically chosen to match the resource restrictions (latency, size) on mobile devices for this application. To evaluate the performance of DFU localization using heavy model, we also include R-FCN with ResNet101 to our experiment.

Inception-V2 is a new iteration of the original inception architecture called GoogleNet with new features such as factorization of bigger convolution kernels to multiple smaller convolution kernels and improved normalization. For the first time, this network used depth-wise separable convolutions to reduce the computations in the first few layers. They also in-

TABLE I
PERFORMANCE OF STATE-OF-THE-ART OBJECT LOCALIZATION MODELS ON MS-COCO DATASET [38]

Model Name	*Speed (ms)*	*Size of Model (MB)*	*COCO mAP*
SSD-MobileNet	30	29.2	21
SSD-InceptionV2	42	102.2	24
Faster R-CNN with InceptionV2	57.2	58	28
R-FCN with ResNet101	92	218.3	30
Faster R-CNN with ResNet101	106	196.9	32
Faster R-CNN with Inception-ResnetV2	620	247.5	37

troduced batch normalization layer which can decrease internal covariate shift, also combat the gradient vanishing problem to improve the convergence during training [44].

MobileNet is a recent lightweight CNN which uses depth-wise separable convolutions to build small, low latency models with a reasonable amount of accuracy that matches the limited resource on mobile devices. The basic block of depth-wise separable convolution consists of depth-wise convolution and point-wise convolution. The 3×3 depth-wise convolution is used to apply a single filter per each input channel whereas pointwise convolution is just simple 1×1 convolution used to create the linear combination of the depth-wise convolution output. Also, it uses both batchnorm layers as well as RELU layers after both layers [43].

ResNet101 is one of the residual learning networks which won the first place on ILSVRC 2015 classification task [45]. As suggested by the name, ResNet101 is a very deep network consists of 101 layers which is about 5 times much deeper than VGG nets but still having lower complexity. The core idea of ResNet is providing shortcut connection between layers, which make it safe to train very deep network to gain maximal representation power without worrying about the degradation problem, i.e., learning difficulties introduced by deep layers.

D. The Transfer Learning Approach

CNNs requires a considerable dataset to learn the features to get the positive results for detection of objects in images [5]. It is vital to use transfer learning from massive datasets in non-medical backgrounds such as ImageNet and MS-COCO dataset to converge the weights associated with each convolutional layers of network [10], [48], [49] for training the limited dataset. The main reason for using two-tier transfer learning in this work is because, the medical imaging datasets are very limited. Hence, when CNNs are trained from scratch on these datasets, they do not produce useful results. There are two types of transfer learning i.e. partial transfer learning in which only the features from few convolutional layers are transferred and full transfer learning in which features are transferred from all the layers of previous pre-trained models. We used both types of transfer learning known as two-tier transfer learning [10]. In the first tier, we used partial transfer learning by transferring the features only from the convolutional layers trained on most significant classification challenge dataset called ImageNet which consists of more than 1.5 million images with 1000 classes [37]. In the second tier, we used full transfer learning to transfer the features from a model trained on object localization dataset called MS-COCO that consists of more than 80000 images with 90 classes [38]. Hence, we used the two-tier transfer learning technique to produce the pre-trained model for all frameworks in our DFU localization task.

E. Performance Measures of Deep Learning Methods

We used four performance metrics i.e. *Speed*, *Size of the model*, *mean average precision (mAP)*, and *Overlap Percentage*. The *Speed* determines the time model takes to perform inference on single image whereas *Size of the model* is the total size of the frozen model that is used for the inference of test images. These are crucial factors for the real-time prediction on mobile platforms. The *mAP* has an "overlap criterion" of intersection-over-union greater than 0.5. The *mAP* is an important performance metric extensively used for the evaluation of the object localization task. The prediction by model to be considered a correct detection, the area of overlap A_o between the bounding box of prediction B_p and bounding box of ground truth B_g must exceed 0.5 (50%) [50]. The last evaluation metric is called *Overlap Percentage*, which is mean average of intersection over union for all correct detection.

$$A_o = \frac{area(B_p \cap B_g)}{area(B_p \cup B_g)} \tag{1}$$

III. EXPERIMENT AND RESULT

As mentioned previously, we used the deep learning models based on three meta-architectures for the DFU localization task. Tensorflow object detection API [47] provides an open source framework which makes very convenient to design and build various object localization models. The experiments were carried out on the DFU dataset and evaluated with 5-fold cross-validation technique. First, we randomly split the whole dataset into 5 testing sets (20% each) for 5-fold cross validation. This is to ensure that the whole dataset was evaluated on testing sets. For each testing set (20%), the remaining images was randomly split into 70% for training set and 10% validation set. Hence,

for each fold, we divided the whole dataset of 1775 images into approximately 1242 images in training set, 178 in validation set and 355 in testing set. This was repeated for 5-fold to ensure the whole dataset was included in testing set.

a) Configuration of GPU Machine for Experiments: (1) Hardware: CPU - Intel i7–6700 @ 4.00 Ghz, GPU - NVIDIA TITAN X 12 GB, RAM - 32 GB DDR4 (2) Software: Tensor-flow [47].

We tested four state-of-the-art deep convolutional networks for our proposed object localization task as described in Section III B. We trained the models with input-size of 640 $\times$ 640 using stochastic gradient descent with different learning rate on Nvidia GeForce GTX TITAN X card. We initialised the network with pre-trained weights using transfer learning rather than randomly initialized weights for the better convergence of the network. We tested the multiple learning rates by decreasing the original learning rates with the 10 and 100 times as well as multiplication factor from 1 to 5 to check the overall minimal validation loss. For example, if the original Inception-V2 learning rate was set at 0.001. Then, for training on DFU dataset, we used 10 learning rates of 0.0001, 0.0002, 0.0003, 0.0004, 0.0005, 0.00001, 0.00002, 0.00003, 0.00004, 0.00005.

We used 100 epochs for training of each reported model, which we found are sufficient to train the DFU dataset as both training and validation loss finally converge to optimal lowest. We selected the models on the basis of minimum validation losses for the evaluation. We tried different hyper-parameters such as learning rate, number of steps and data augmentation options for each model to minimize both training and validation losses. In next section, we report the different network hyper-parameters and configurations for each model used for evaluation on the DFU dataset.

We set the appropriate hyper-parameters on the basis of meta-architecture to train the models on DFU dataset. For SSD, we used two CNNs, MobileNet and Inception-V2 (both of them use depth-wise separable convolutions), we set the weight for `l2_regularizer` as 0.00004, initializer that generates a truncated normal distribution with standard deviation of 0.03 and mean of 0.0, `batch_norm` with decay of 0.9997 and epsilon of 0.001. For training, we used a batch size of 24, optimizer as `RMS_Prop` with a learning rate of 0.004 and decay factor of 0.95. The momentum optimizer value is set at 0.9 with a decay of 0.9 and epsilon of 0.1. We also used two types of data augmentation as random horizontal flip and random crop. For Faster-RCNN, we set the weight for `l2_regularizer` as 0.0, initializer that generates a truncated normal distribution with standard deviation of 0.01, `batch_norm` with decay of 0.9997 and epsilon of 0.001. For training, we used a batch size of 2, optimizer as momentum with manual step learning rate with an initial rate as 0.0002, 0.00002 at epoch 40 and 0.000002 at epoch 60. The momentum optimizer value is set at 0.9. For training RFCN, we used same hyper-parameters as Faster-RCNN with only change in the learning rate set as 0.0005. For data augmentation, we used only random horizontal flip for these two meta-architectures.

In Table II, we report the performance evaluation of object localization networks for DFU dataset on 5-fold cross validation. Overall, all the models achieved promising localization results with high confidence on DFU dataset. Few instances of accurate localization by all trained models are demonstrated by the Fig. 11. SSD-MobileNet ranked first in the *Size of Model* and *Average Speed* performance index. This is mainly due to the simpler architecture to generate anchor boxes in SSD [42]. Whereas in *Ulcer mAP* and *Overlap Percentage*, R-FCN with ResNet101 and Faster R-CNN with InceptionV2 were almost equally competitive in these performance measures. In *Ulcer mAP*, Faster R-CNN with InceptionV2 ranked first with overall mAP of 91.8%, just slightly better than R-FCN with ResNet101 with mAP of 90.6%. But, in *Overlap Percentage*, R-FCN-Resnet101 achieved a score of 96.1%, which was slightly better than Faster R-CNN with Inception. SSD-InceptionV2 ranked third in both of these performance measure categories with difference of 4.6% in *Ulcer mAP* and 3.5% in *Overlap Percentage* from the first position. In performance measures, overall Faster R-CNN with InceptionV2 was the best performer, and the most lightweight SSD-MobileNet emerged as the worst performer in terms of accuracy. Finally, we tested models on the dataset of 105 healthy foot images for specificity measure. None of the above-mentioned models produce any DFU localization on these healthy images.

A. Inaccurate DFU Localization Cases

In this work, we explored different object localization meta-architectures to localize DFU on full foot images. Although the performance of all models is quite accurate as shown in the Fig. 11, this section explores inaccurate localization cases by trained models on DFU dataset in 5-fold cross-validation as shown in the Fig. 12. We found that trained models were struggled to localize the DFU of very small size and that has the similar skin tone of the foot especially, SSD-MobileNet and SSD-InceptionV2. There are cases of DFU that have very subtle features, not even, most accurate models such as Faster-RCNN with InceptionV2 and R-FCN with ResNet101 were able to detect these conditions.

IV. Inference of Trained Models on NVIDIA Jetson TX2 Developer Kit

Nvidia Jetson TX2 is the latest mobile computer hardware with an onboard 5-megapixel camera and a GPU card for the remote deep learning applications as shown in the Fig. 13. However, it is not capable of training large deep learning models. We installed tensor-flow specifically designed for this hardware to produce inference from the DFU localization models that we trained on the GPU machine. Jetson TX2 is a very compact and portable device that can be used in various remote locations.

b) Configuration of Jetson TX2 for Inference: (1) Hardware: CPU - dual-core NVIDIA Denver2 + quad-core ARM Cortex-A57, GPU - 256-core Pascal GPU, RAM - 8 GB LPDDR4 (2) Software: Ubuntu Linux 16.04 & Tensor-flow.

We did not find any difference in the prediction of the models on Jetson TX2 hardware and the GPU machine; the only let-off is the slow inference speed on the Jetson TX2. It is obviously

TABLE II
PERFORMANCE MEASURES OF OBJECT LOCALIZATION MODELS ON THE DFU DATASET

Model Name	*Speed (ms)*	*Size of Model (MB)*	*Ulcer mAP*	*Overlap Percentage (%)*
SSD-MobileNet	28	22.6	84.9	89.4
SSD-InceptionV2	37	53.5	87.2	92.6
Faster R-CNN with InceptionV2	48	52.2	91.8	95.8
R-FCN with Resnet 101	90	199.1	90.6	96.1

Fig. 11. The accurate localization results to visually compare the performance of object localization networks on the DFU dataset. Where SSD-MobNet is SSD-MobileNet, SSD-IncV2 is SSD-InceptionV2, FRCNN-IncV2 is Faster R-CNN with InceptionV2, and RFCN-Res101 is R-FCN with ResNet101.

due to limited hardware compared to the GPU machine. For example, the speed of SSD-MobileNet was 70 ms per inference on Jetson TX2 as compared to 30 ms on GPU machine. Also, for real-time localization, models can produce the visualization of maximum 5 fps using the on-board camera with lightweight model. Fig. 14 demonstrates the inference using Jetson TX2.

V. REAL-TIME DFU LOCALIZATION WITH SMARTPHONE APPLICATION

Training and inference of the deep learning frameworks on smartphone are challenging tasks due to limited resources of a smartphone. Hence, we trained these object localization frameworks on the desktop with a GPU card. We utilized the

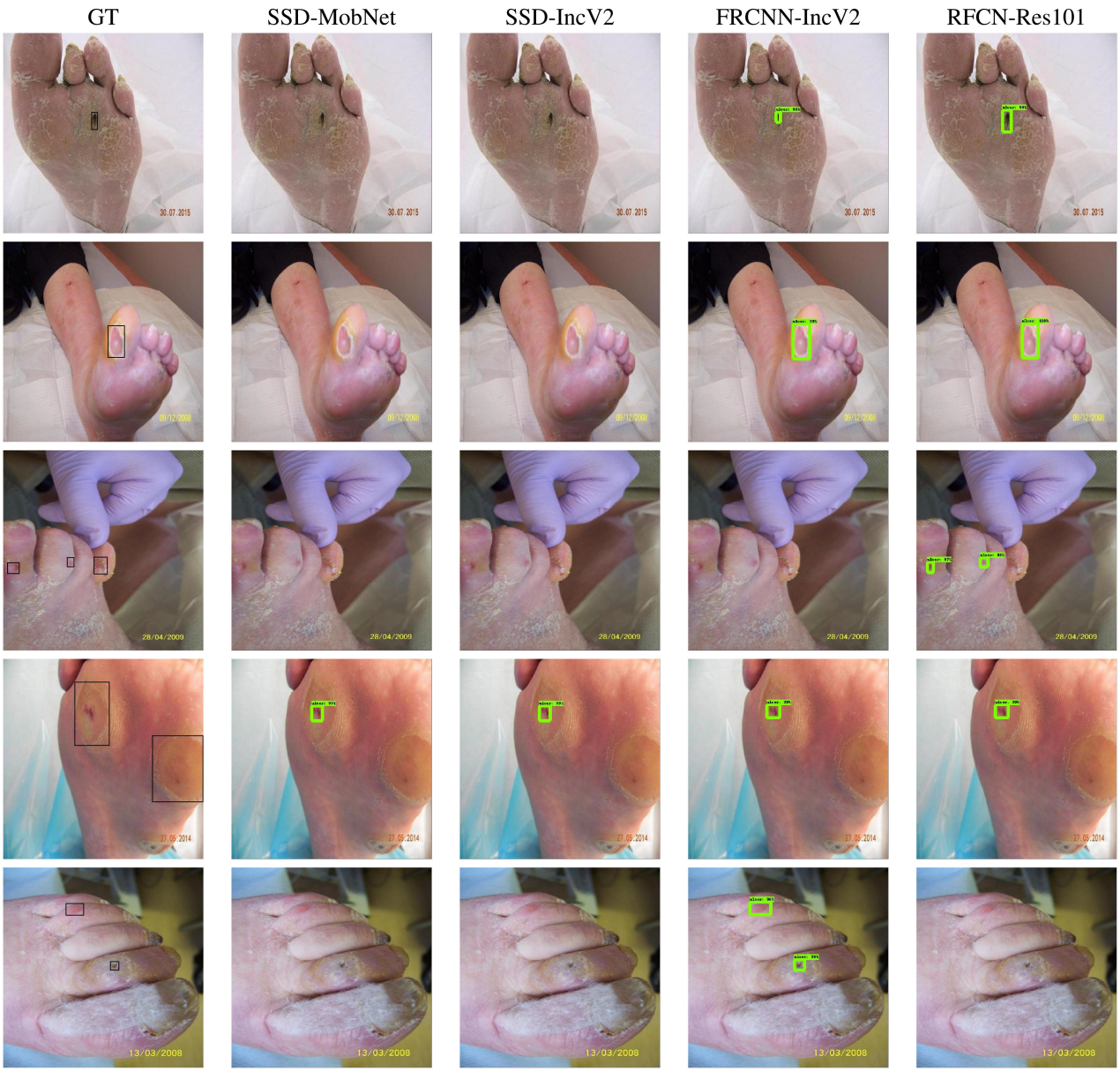

Fig. 12. Incorrect localization results to visually compare the performance of object localization networks on DFU dataset. Where SSD-MobNet is SSD-MobileNet, SSD-IncV2 is SSD-InceptionV2, FRCNN-IncV2 is Faster R-CNN with InceptionV2, and RFCN-Res101 is R-FCN with ResNet101.

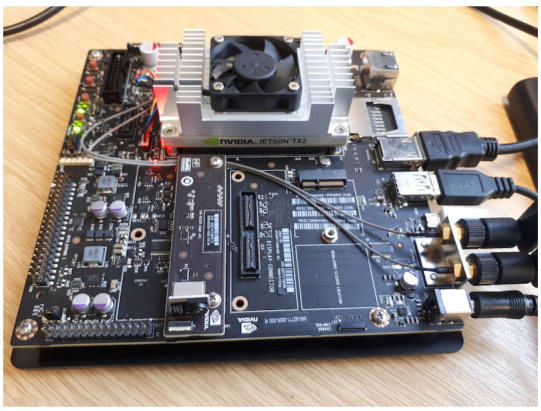

Fig. 13. Nvidia Jetson TX2.

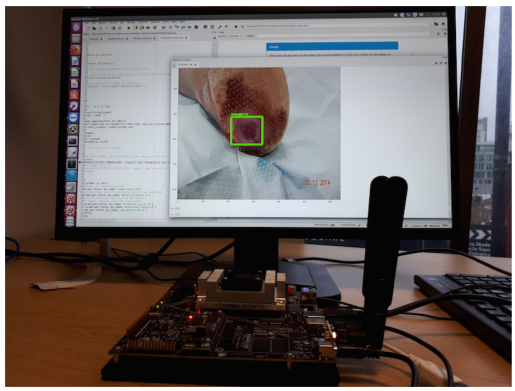

Fig. 14. DFU localization on Nvidia Jetson TX2 using Faster R-CNN with InceptionV2 on tensor-flow.

Fig. 15. Real-time localization using smartphone android application. In the first row, images are captured by default camera. In the second row, the snapshot of real-time localization by our prototype android application.

whole dataset of 1775 DFU images for further experiments by randomly splitting 90% data in the training set and remaining 10% in the validation set. We trained only Faster R-CNN with InceptionV2 on this dataset because of the best trade-off between the accuracy and the speed. With android studio and tensor-flow deep learning mobile library, we deployed these models on Samsung A5 2017 (Android Phone) to create the real-time object localization for DFU. As mentioned in the previous section, we finalized Faster R-CNN with InceptionV2 model for the prototype android application.

We tested our prototype application for the real-time application in real-time healthcare settings as shown in the Fig. 15. We tested this application on 30 people in this preliminary test in which 10 people were with DFU. Out of 10 people with DFU, our application detected 8 DFU and out of 20 people with normal foot, our application did not detect any false detection. Furthermore, more user-friendly features, care, and guidance will be added to this application to make it a complete package of DFU care for diabetic patients.

VI. Discussion and Conclusion

Diagnosis and detection of DFU by the computerized method has been an emerging research area with the evolution of computer vision, especially deep learning methods. In this work, we investigated the use of both conventional machine learning and deep learning for the DFU localization task. We achieved relatively good performance using conventional machine learning technique. But, due to multiple intermediate steps, this approach is very slow for the DFU localization task. In deep learning, we used different object localization meta-architectures to train the end-to-end models on the DFU dataset with different hyper-parameter settings and two-tier transfer learning to localize DFU on the full foot images with high accuracy. As shown in the Fig. 11, these methods are capable of localizing multiple DFU with high inference speed. We also found that though SSD meta-architecture produced fastest inference due to the two-stage architecture, Faster R-CNN produced the most accurate results in our task. Then, we demonstrated how these methods can be easily transferred to a portable device, Nvidia Jetson TX2, to produce inference remotely. Finally, these deep learning methods were used in an android application to provide real-time DFU localization. In this work, we developed mobile systems that can assist both medical experts and patients for the DFU diagnosis and follow-up in the remote settings.

In the present situation, manual inspection by podiatrists remains the ideal solution for the diagnosis of DFU. However, Netten *et al.* [51] claimed that human observers achieved low validity and reliability for remote assessment of DFU. Therefore, computerized method could be used as a tool to improve human performance. Developing the remote, computerized and innovative DFU diagnosis system according to the medical classification systems and exactness accomplished by the podiatrist, it demands a significant amount of research. To assist podiatrist, foot analysis with computerized methods in the near future, the following issues need to be addressed.

1) The detection of DFU on foot images with computerized methods is a difficult task due to high inter-class similarities and intra-class variations in terms of color, size, shape, texture and site amongst different classes of DFU. Although, detection and localization of DFU on full foot images is a valuable study, further analysis of each

DFU on foot images is required according to the medical classification systems followed by podiatrists such as the Texas Classification of DFU [29] and the SINBAD Classification System [52]. Most of the state-of-the-art computerized imaging methods rely on the supervised learning. Hence, there is a need for laborious manual annotation by medical experts according to these popular classification systems. For example, the Texas classification system classifies DFU into 16 classes depending on conditions of DFU based on ischemia, infection, area and depth. These methods can be extended to produce localization of DFU and determine the outcome of DFU according to the Texas classification system with substantial image data belonging to each class and expert annotations.

2) Deep learning methods require a considerable amount of data to learn features of abnormality in medical imaging. To achieve accurate DFU detection according to different classification systems, multiple images of same DFU covering key specific conditions such as lighting conditions, the distance of image capture from the foot and orientation of the camera relative to the foot. To our best knowledge, there are no publicly available standardized DFU dataset with descriptions and annotation. Hence, there is a requirement for a publicly available annotated DFU dataset with essential diagnostic capability in this regard. The standardized dataset can help to produce even more accurate results with these methods.
3) Early detection of key pathological changes in the diabetic foot leading to the development of a DFU is really important. Hence, the time-line dataset of patients with early signs of DFU till the diagnosis is required to achieve this objective. With these methods and time-line dataset, the early prediction, healing progress and other potential outcomes of DFU could be possible.
4) The combination of image features and diagnosis features such as patient's ethnicity, the presence of ischemia, depth of DFU to the tendon, neuropathy would aid to a more robust DFU diagnosis system.
5) The DFU diagnosis system should be scalable to multiple devices, platforms and operating systems.

With limited human resources and facilities in healthcare systems, DFU diagnosis is a significant workload and burden for the government. The computer-based systems have huge potential to assist healthcare systems in DFU assessment. The new technologies like the Internet of Things (IoT), cloud computing, computer vision and deep learning can enable computer systems to remotely assess the wounds, provide faster feedback with good accuracy. But, this integrated system should be tested and validated rigorously by podiatrists and medical experts, before it is implemented in the real healthcare setting and deployed as a mobile application.

Acknowledgment

The authors express their gratitude to Lancashire Teaching Hospitals and J. Spragg who is a Podiatrist/Chiropodist in the Rossendale Practice, Rawtenstall, Lancashire for their extensive support and contribution in carrying out this research. They gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPU used for this research.

References

[1
[2
[3
[4
[5
[6
[7
[8
[9
[10
[11
[12
[13
[14
[15
[16
[17
[18
[19
[20
[21

[22
[23
[24
[25
[26
[27
[28
[29
[30
[31
[32
[33
[34
[35
[36
[37
[38
[39
[40
[41
[42
[43
[44
[45
[46
[47
[48
[49
[50
[51
[52